

TITLE OF THE INVENTION
CODE PROCESSING CIRCUIT
BACKGROUND OF THE INVENTION

1 Field of the Invention

5 本発明は、符号を処理する符号処理回路に関する。

2 Description of the Related Art

符号処理回路では、複数の符号化されたデータを順に出力する場合、出力するビット数に符号長を合わせこんで出力しなければならなかつた。つまり、8ビット（1バイト）単位で出力する場合、符号化されたデータの符号長が27ビットであれば、8ビット単位にするために5ビットのダミーデータを附加して出力していた。

10 このように出力するデータに複数の符号化されたデータが存在する場合、符号化することでデータ長を短くしているにも関わらず、それぞれの符号を合わせ込む処理を簡易化するために、符号を出力ビット数で割り切れるデータ長にダミーデータを附加して合わせこむ。そのため、符号化したデータ長よりも出力するデータのデータ長が長くなつてしまつ
15 ている。また、このようなデータは、符号の打ち切りを行う場合に、各符号長がわからなくなつたため、最適な符号の打ち切りができなかつた。

したがつて、符号を出力ビット単位で合わせこんで出力することができる符号処理回路に対するneedがある。

BRIEF SUMMARY OF THE INVENTION

20 本発明の一態様によると、符号処理装置は、複数の種類のデータをそれぞれ符号化する複数の符号部と、前記複数の符号部と対応して設けられた前記符号部から出力された符号を記憶する第1のバッファと、前記複数の符号部と対応して設けられた前記符号部から出力された符号長を記憶する第2のバッファと、前記複数の符号部と対応して設けられた前記第2のバッファに記憶する符号長を加算する第1の加算部と、前記各第1の加算部で加算された符号長の全てを加算する第2の加算部と、前記第1のバッファに記憶された符号、前記第2のバッファに記憶された符号長及び前記第2の加算部で加算された符号長に基づいて出力符号を1ビット単位で合わせこむ合わせこみ部とを具備する。

30 Objects and advantages of the invention will become apparent from the description which follows, or may be learned by practice of the invention.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWING

The accompanying drawings illustrate embodiments of the invention, and together with the general description given above and the detailed description given below, serve to explain the principles of the invention.

5 図1は、本発明の一実施例における符号処理回路の構成を示す図である。

図2は、出力符号作成部の動作イメージを示すタイミングチャートである。

図3は、符号長記憶部に記憶されるビット数の一例を示す図である。

図4は、ビット単位での合わせこみを説明するための図である。

10 図5は、符号長記憶部に記憶されるビット数の他の一例を示す図である。

図6は、打切りを行うビットを示す図である。

図7は、打切りを行った後のビット単位での合わせこみを説明するための図である。

DETAILED DESCRIPTION OF THE INVENTION

15 以下、本発明の一実施例について図面を参照して説明する。この一実施例では、符号処理回路は、入力される赤（R）緑（G）青（B）データを符号化する符号部が各色毎に3つ設けられており、符号化したデータを8ビット（1バイト）単位で出力する場合で説明する。

符号処理回路100は、図1に示すように構成されている。

20 符号処理回路100は、Rデータを符号化する回路である第1の符号器1、第1の符号器1で符号化されたR符号を記憶するR符号記憶回路であるバッファ2、前記符号化されたR符号のデータ長であるR符号長を記憶するR符号長記憶回路であるバッファ3、前記R符号長を加算するR符号長加算回路である第1の加算器4を有している。このような構成は、Gデータ、Bデータに対しても同様に設けられている。すなわち、Gデータに対して第2の符号器5、G符号を記憶するバッファ6、G符号長を記憶するバッファ7及び第2の加算器が8設けられ、Bデータに対して第3の符号器9、B符号を記憶するバッファ10、B符号長を記憶するバッファ11及び第3の加算器12が設けられている。なお、各符号器1、5及び9にはデータサイズは同じだが別々のデータが順次入力されるため、それぞれの符号、符号長は偶然を除き全く異なるものとなっている。

30 また、符号処理回路100は、さらに第4の加算器13、比率部14、合わせこみ部1

5を具備している。

第4の加算器13は、各加算器4, 8及び12で加算されたR符号長、G符号長、B符号長の全ての符号長を加算して合計加算値を算出するRGB符号長加算回路である。

前記比率部14は、R符号長、G符号長及びB符号長の比率を出力する回路である。比率部14は比率算出部14a、比率設定部14b、切換部であるセレクタ14c、使用比率選択部14dとを具備している。比率算出部14aは、各加算器4, 8及び12で加算された符号長及び第4の加算器13で加算された符号長とからRGB各色の符号長の比率を算出する。比率設定部14bには、RGB各色の符号長の比率が予め設定される。セレクタ14cは、前記比率算出部14aで算出された比率と前記比率設定部14bに設定された比率とのいずれかを切換えて出力する。使用比率選択部14dは、例えば、符号処理回路100が設けられた装置の外部のスイッチのON/OFFに基づいて、セレクタ14cの切換えを制御して比率算出部14aに設定された比率を有効にするか、比率設定部14bに設定された比率を有効にするかを決定する。

合わせこみ部15は、バッファ2, 6及び10に記憶された符号、バッファ3, 7及び11に記憶された符号長及び第4の加算器13で加算された符号長等に基づいて出力符号データを1ビット単位で合わせこむ回路である。合わせこみ部15は、指定符号長記憶部15a、符号長比較部15b、イネーブル信号作成部15c、出力符号作成部15dとを具備している。

符号長比較部15bは、出力する出力符号データの符号長の許容符号長を指定符号長として設定する指定符号長記憶部15aに記憶された符号長の単位と前記第4の加算器13で加算された合計加算値とを比較する。イネーブル信号作成部15cは、各加算器4, 8及び12で加算されたRGB各色の符号長、前記比率部14で出力されたRGB符号長の比率及び符号長比較部15bで比較された比較結果に基づいて各色符号のいずれが有効であるかを示す有効符号信号を作成する。出力符号作成部15dは、前記各RGB符号およびイネーブル信号作成部15cで作成された有効符号信号とから出力符号データを作成する。

以下、このように構成された符号処理回路100のデータの処理の流れを説明する。まず、Rデータは、第1の符号器1に入力されると符号化される。そして、第1の符号器1から出力された符号と符号長は、それぞれバッファ2、バッファ3に格納される。このときそれぞれのバッファ2, 3に入力されるR符号データをR符号、そのR符号データのデ

一タ長をR符号長とする。R符号長はバッファ3への格納と並行して第1の加算器4に入力され、第1の加算器4で加算される。この加算されたR符号長をR加算値とする。続いて、Gデータに対しても同様な動作が行われる。このとき第2の符号器5から出力される符号をG符号、そのG符号のG符号データの長さをG符号長、第2の加算器8で加算されるG符号長をG加算値とする。さらに、Bデータに対しても同様な動作が行われる。このとき、第3の符号器9から出力される符号をB符号、そのB符号のB符号データの長さをB符号長、第3の加算器12で加算されるB符号長をB加算値とする。

そして、各加算器4, 8及び12は、全ての各RGB符号長の全ての長さを把握するためにR加算値、G加算値、B加算値を第4の加算器13及び比率算出部14aへ出力する。この第4の加算器13により入力されたRGBデータ全体の符号長を把握することができる。この第4の加算器13で加算された各RGB符号長の加算値の合計を合計加算値とする。この合計加算値は、比率算出部14aへ出力される。

比率算出部14aでは、R符号長、G符号長及びB符号長の比率を、R加算値、G加算値、B加算値及び合計加算値から算出する。例えば、(R加算値／合計加算値) : (G加算値／合計加算値) : (B加算値／合計加算値)によりそれぞれの符号長の比率を算出する。このように算出された比率はセレクタ14cへ出力される。また、上述した比率設定部14bに設定されたRGB符号長の比率もセレクタ14cへ出力される。比率設定部14bの比率の設定は符号処理回路100が設けられた装置の操作部等の指定によって指定されるようにしても良いし、予め複数の比率が設定してあり前記操作部等の指定によって比率が指定されるようにしても良い。このように、セレクタ14cには2種類の比率が入力される。セレクタ14cからは、前記使用比率選択部14dの選択に基づいていずれか一方の比率が合わせこみ部15へ出力される。

続いて、符号長比較部15bで第4の加算器13から入力された合計加算値と指定符号長記憶部15aに記憶された指定符号長の値とを比較する。合計加算値が指定符号長の値以下であればRGB各符号はビットの打切りを行わず、合計加算値が指定符号長の値より大きければRGB各符号を指定符号長の値以下となるようにビットの打切りを行う。このRGB各符号の打切りの方法については後述する。

そして、イネーブル信号作成部15cで比率部14から入力された比率、符号長比較の結果、バッファ3, 7及び11から入力される符号長を基に、各色符号のいずれが有効であるかを示す有効符号信号であるRイネーブル信号、Gイネーブル信号及びBイネーブル

信号を作成する。そして、出力符号作成部 15 d はイネーブル信号作成部 15 c で作成された各色のイネーブル信号を元に出力符号データを作成する。図 2 は、出力符号作成部 15 d の動作イメージの一部を示すタイミングチャートである。

図 2 中、各色の符号は 16 ビットの各色の符号データを示し、各色のイネーブルは RG B の符号に対応する 16 ビットの信号を示している。各色のイネーブル信号が “0” を示すときにその色に対応する色の符号が有効になるようになっている。バッファイネーブルは、有効なバッファがバッファ 2, 6 及び 10 のいずれであるかを示す信号である。また、レジスタ R 1 は、有効バッファの符号データの上位 8 ビットを記憶し、レジスタ R 2 は、有効バッファの符号データの下位 6 ビットを記憶し、レジスタ R 3 は、符号データの有効符号を記憶する。また、出力符号は、8 ビットの出力符号データを示している。

同図に示すように、始めは、バッファイネーブル信号がバッファ 2 を選択しているので、バッファ 2 から R 符号を読み込み、その上位 8 ビットをレジスタ R 1 に、下位 6 ビットをレジスタ R 2 に入れる。読み込んだ R 符号に対応した R イネーブル信号から、R 符号の有効符号をレジスタ R 3 に入れる。そして、バッファイネーブル信号がバッファ 6 に変化したら、有効なバッファがバッファ 2 からバッファ 6 に変るので、バッファ 6 から B 符号を読み込みこむ。以下 R 符号を読み込む場合と同様な処理を行い出力符号を作成していく。

続いて、合わせこみ部 15 での処理について説明する。合わせこみ部 15 は、RGB 各符号を、RGB 各符号長、合計加算値及び前記比率に基づいて合わせこむタイミングは、例えば、次のタイミングで行う。各符号器 1, 5 及び 9 から符号化終了フラグ（R 符号化終了信号、G 符号化終了信号、B 符号化終了信号）を通知してもらい、全てのフラグが通知されたときに、合わせこみ部 15 は RGB 各符号の合わせこみを行う。

次に、図 3 から図 7 を参照して各色の符号の合わせこみをビット単位で行う方法について模式的に説明する。

符号化終了フラグが全て通知されたときに、符号長比較部 15 b により、第 4 の加算器 13 から入力された合計加算値と指定符号長記憶部 15 a に記憶された指定符号長の値とが比較される。

先ず、図 3 に示すように、指定符号長記憶部 15 a に 560 ビットが記憶されており、合計加算値が 560 ビット以内の場合を想定する。合計加算値が 560 ビット以内の例として、図 4 に示すように、R 符号 180 ビット、G 符号 60 ビット、B 符号 310 ビットの 550 ビットの場合を挙げる。

この場合は、合計加算値が 560 ビット以内なので、3つの符号長をそのまま出力すれば良い。しかしながら出力符号データの出力単位はバイト単位であることを考慮する必要がある。そこで、各色の符号長の加算値がわかっているので、R 加算値 180 を 8 で除算する。除算結果から R 符号はバイト単位にすると 4 ビットの余りが発生してしまうので、
5 G 符号より 4 ビットのデータを付加して出力する。続いて、G 加算値 60 から R 符号に付加した 4 ビットを減算した 56 を 8 で除算する。この場合は割り切れるので、G 符号の最後の符号データは 1 バイトの 8 番目のデータとなる。このように、ビット単位でダミーデータを付加せずに出力符号データを作成していくことができる。

また、図 5 に示すように、指定符号長記憶部 15a に 480 ビットが記憶されていたとすると、図 4 で示した R G B 各色の符号のように合計加算値が 550 ビットの場合は、70 ビット以上の符号の打切りを行わなくてはならない。この場合は先ず打切りビット数が算出される。このビット数の打切りには比率部 14 から出力された比率を用いる。比率部 14 から入力された比率が比率算出部 14a で算出された比率の場合は、各色の符号長の比は、上述した算出式を用いて算出すると、略 $R : G : B = 3 : 1 : 5$ となっている。したがって、R 符号の打切りビット数の目安は略 24 ビット ($= 70 / 9 * 3$) 、G 符号の打切りビット数の目安は略 8 ビット ($= 70 / 9 * 1$) 、B 符号の打切りビット数の目安は略 39 ビット ($= 70 / 9 * 5$) となる。

R G B 各色の符号が図 6 で示すように構成されていた場合、前記算出した打切りビット数を目安とし、70 ビット以上打切るようにビットが選択される。R 符号は、N 番目 (10 ビット) 、N-1 番目 (4 ビット) 、N-2 番目 (9 ビット) が選択され、23 ビットを打切りビットとする。G 符号は、N 番目 (7 ビット) が選択され、7 ビットを打ち切りビットとする。B 符号は、N 番目 (3 ビット) 、N-1 番目 (8 ビット) 、N-2 番目 (4 ビット) 、N-3 番目 (8 ビット) 、N-4 番目 (5 ビット) 、N-5 番目 (12 ビット) が選択され、40 ビットを打切りビット数とする。これらの打切りビット数を加算すると、70 ビットとなり、打切りビットを 70 以上とすることができます。

したがって、出力する R 符号は 157 ビット、G 符号は 53 ビット、B 符号は 270 ビットとなる。符号処理回路 100 は、このように、符号一つ一つの符号長を把握しているため、最適な打切りを行うことができる。また、このように打切りが行われた後の R G B 各色の出力符号のビット単位の合わせこみは、図 7 に示すように行われる。

30 R 符号長 157 を 8 で除算すると、5 ビットの余りが発生してしまうので、G 符号より

3ビットのデータを付加して1バイト単位にして出力する。そして、G符号53ビットからR符号に付加した3ビットを減算した50を8で除算すると、2ビットの余りが発生してしまう。そこで、B符号から6ビットのデータを付加して出力する。これにより、指定された符号長より長い合計加算値であった場合にビットの打切りを行った場合にも、ビット単位でダミーデータを付加せずに出力符号データを作成していくことができる。

また、B符号長を8で除算したときに余りが発生する場合には、その余りのビットに応じたビットを、バッファ2に記憶されている次のR符号から付加する。次のR符号とは、次のRGBデータのブロックのRデータが第1の符号器1で符号化されたものである。このように構成すると、RGBデータのブロック間においても、ダミーデータを付加しないようにすることができる。これにより、例えば、カラー画像を読取るスキャナに符号処理回路100を設けた場合に、大量に読込まれるRGBデータのブロックを処理するときにダミーデータを付加せずに出力することができるので、RGB符号間だけでなく、RGBデータのブロック間にもダミーデータを付加しないので出力符号データを大幅に低減できる。

なお、この実施の形態では、複数の種類のデータとして、RGBデータの場合で説明しているがこれに限られるものではなく、RGB以外のデータでも、複数の種類のデータを用いる場合に適用できる。

Additional advantages and modifications will readily occur to those skilled in the art. Therefore, the invention in its broader aspects is not limited to the specific details and representative embodiments shown and described herein. Accordingly, various modifications may be made without departing from the spirit or scope of the invention as defined by the appended claims and equivalents thereof.

WHAT IS CLAIMED IS:

1. 符号処理回路, comprising:

複数の種類のデータをそれぞれ符号化する複数の符号部と、

前記複数の符号部と対応して設けられた前記符号部から出力された符号を記憶する第1

5 のバッファと、

前記複数の符号部と対応して設けられた前記符号部から出力された符号長を記憶する第2のバッファと、

前記複数の符号部と対応して設けられた前記第2のバッファに記憶する符号長を加算する第1の加算部と、

10 前記各第1の加算部で加算された符号長の全てを加算する第2の加算部と、

前記第1のバッファに記憶された符号、前記第2のバッファに記憶された符号長及び前記第2の加算部で加算された符号長に基づいて出力符号を1ビット単位で合わせこむ合わせこみ部。

15 2. 符号処理回路, according to claim 1, 前記合わせこみ部は、出力する符号長の単位を記憶する符号長記憶部と、前記第2の加算部で加算された符号長と符号長記憶部に記憶された符号長とを比較する符号長比較部と、前記第2の記憶部に記憶された符号長及び前記符号長比較部で比較された比較結果に基づいて前記複数の種類の有効符号信号を作成するイネーブル信号作成部と、前記第1の記憶部に記憶された符号および前記イネーブル信号作成部で作成された有効符号信号とから出力符号を1ビット単位で作成する出力符号作成部を具備する。

20 3. 符号処理回路 according to claim 1, 前記複数の種類のデータの出力する符号の比率を算出する比率部をさらに具備し、前記合わせこみ部は、前記符号長記憶部に記憶された出力単位より第2の加算部で加算された符号長の値が大きいときは、前記比率に基づいて符号の打切りを行う。

25 4. 符号処理回路 according to claim 3, 前記比率部は、前記各第1の加算部で加算された符号長及び前記第2の加算部で加算された符号長とから各符号長の比率を算出する比率算出部を具備する。

5. 符号処理回路 according to claim 3 前記比率部は、前記第1の加算部で加算された各符号長の比率を予め設定する比率設定部を具備する。

30 6. 符号処理回路 according to claim 4, wherein 前記比率部は、前記第1の加算部

で加算された各符号長の比率を予め設定する比率設定部と、前記比率設定部に設定された比率と前記比率算出部で算出された比率とのいずれかを切換えて出力する切換部を具備する。

7. 符号処理回路 according to claim 1, wherein 前記複数の種類のデータは、赤データ、緑データ、青データである。

5

ABSTRACT OF THE DISCLOSURE

符号処理回路は、複数の種類のデータをそれぞれ符号化する複数の符号部と、前記複数の符号部と対応して設けられた前記符号部から出力された符号を記憶する第1のバッファと、前記複数の符号部と対応して設けられた前記符号部から出力された符号長を記憶する第2のバッファと、前記複数の符号部と対応して設けられた前記第2のバッファに記憶する符号長を加算する第1の加算部と、前記各第1の加算部で加算された符号長の全てを加算する第2の加算部と、前記第1のバッファに記憶された符号、前記第2のバッファに記憶された符号長及び前記第2の加算部で加算された符号長に基づいて出力符号を1ビット単位で合わせこむ合わせこみ部とを具備する。